AD-A247 003

IMENTATION PAGE

Form Approved
OMB No. 0704-0188

2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
October 1991	Final Report 15 JUL 89 - 14 JUL91

4. TITLE AND SUBTITLE
"Evolutionary Approach to Designing Neural Networks"(U)

5. FUNDING NUMBERS
61102F
2305/B3

6. AUTHOR(S)
Aviv Bergman, Research Physicist

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
SRI International
Artificial Intelligence Center
333 Ravenswood Avenue
Menlo Park, CA 94025

AFOSR-TR- 92 0010

8. PERFORMING ORGANIZATION REPORT NUMBER
SRI Project ECU 7929

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
USAF, AFSC
Air Force Office of Scientific Research
Building 410
Bolling Air Force Base, Washington, DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
F49620-89-K-0005

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for Public Release:
Distribution Unlimited

12b. DISTRIBUTION CODE
UL

13. ABSTRACT *(Maximum 200 words)*

One of the most interesting properties of neural networks is their ability to learn appropriate behavior by being trained on examples. Established learning algorithms, which typically work by minimizing error through backpropagation in weight space, tend to get stuck in local optima--a tendency typical of gradient-descent methods applied to nonconvex objectives functions. Therefore, for problems of nontrivial complexity these systems must be handcrafted to a significant degree, but, the distributed nature of neural network representations make this handcrafting difficult. We are investigating an evolutionary approach to learning that will avoid this problem. This approach simulates a variable population of networks which, through processes of mutation, combination, selection, and differential reproduction, converges to a group of networks well suited to solving the task at hand. The role of the different genetic operation, e.g., recombination and mutation was also studied. We use a Connection Machine to exploit the inherent parallelism in these simulations.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Population Dynamics, Evolution and Coevolution, Unsupervised learning, Adaptation, Neural Networks, Genetic Algorithm.	53
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

Final Report • 31 October 1991

AN EVOLUTIONARY APPROACH TO DESIGNING NEURAL NETWORKS

SRI Project 7929
Contract No. F49620-89-K-0005

Prepared for:

Dr. Steven Suddarth
Air Force Office of Scientific Research, Building 410
Bolling Air Force Base, Washington, DC 20332

Prepared by:

Aviv Bergman, Project Leader
Artificial Intelligence Center
Computer and Information Sciences Division

Approved by:

C. Raymond Perrault, Director
Artificial Intelligence Center
Computing and Engineering Sciences Division

Donald L. Nielson, Vice President and Director
Computing and Engineering Sciences Division

Contents

1 Summary

Research on an evolutionary approach to designing neural networks that learn was begun at SRI International (SRI) in July 1989 under AFOSR sponsorship (SRI Project 7929, Contract No. F49620-89-K0005). This report describes the research conducted during the first two years of the project.

1.1 Objectives

One of the most interesting properties of neural networks is their ability to learn appropriate behavior by being trained on examples. Established learning algorithms, which typically work by minimizing error through backpropagation in weight space, tend to get stuck in local optima — a tendency typical of gradient-descent methods applied to nonconvex objective functions. Therefore, for problems of nontrivial complexity these systems must be handcrafted to a significant degree, but the distributed nature of neural network representations make this handcrafting difficult.

Our goal is to develop a learning and adaptation mechanisms capable of coping with complex and dynamic problem domains. Once we obtain a machine that performs a certain task well, we want to understand *why* its structure leads to good performance, and thereby help a network designer to create even more successful designs. More specifically, the aim of this program is to design a system that can learn to recognize signals adaptively. That is, the system should learn to respond in a distinctive, repeatable way to those signals to which it has been exposed; should track changes to its signal environment (including possibly the introduction of entirely new classes of signals); and should do these things spontaneously, with no instruction. Adaptive signal recognition should be the result of a self-reorganization of the system in the face of a changing environment.

1.2 Approach

We are investigating an evolutionary approach to learning with two levels of representation: genotypic and phenotypic. In this approach a genotype is a highly structured encoding of a class of neural networks, which play the role of phenotypes. The genotype specifies general properties of the networks, such as initial patterns of connectivity, distributions of weights, thresholds or gains, etc. A phenotypic

network can then be further modified to respond appropriately to experienced stimuli (in particular, to classify stimuli).

A fundamental hypothesis motivating this approach is that the principles of biological evolution and population genetics provide the basis for such behavior. The processes of variation and selection, operating at *both* levels of representation, are known to produce in natural populations the kind of emergent behavior we seek to emulate. By simulating these processes on the computer, we observe similar kinds of behavior in artificial systems.

Genotypic variation is caused by random mutation and recombination of the network descriptions; genotypic selection is caused by differential reproduction governed by the performance of networks as measured by an explicit or implicit fitness function. These processes operate over a comparatively long time scale and produce networks with comparatively general adaptations.

Our purpose is not to model biological processes explicitly, but rather to explore a genetic and ecological metaphor of computation. We are interested in investigating this metaphor for two reasons. First of all, adaptive behavior may lead to very general methods of dealing with difficult and ill-defined problems in signal understanding. A system that can learn from experience without explicit training by examples, that can exploit contextual information, and that can modify itself to adapt to possibly radical changes in its input could be useful for difficult problems such as speaker-independent speech recognition. In addition, the inherent parallelism of the evolutionary metaphor, with its emphasis on populations, can lead to effective methods for exploiting the power of parallel computer systems.

We considered two general problems: adaptation to sensory stimuli, and the role genetic operations play in the evolution of learning abilities.

1.3 Summary of Accomplishments

1.3.1 Adaptation Sensory Sensory Stimuli

For the adaptation to sensory stimuli the evolutionary algorithm exhibits effective adaptation, see Appendix A. Differential reproduction amplifies the frequency of selected genes and leads to the emergence of a population that is progressively more fit. In our model, free recombination (crossover)

seems to be the primary means of adaptation. Two relatively fit parents clearly have a better-than-average chance of producing more fit offspring. Mutation, on the other hand, has only an average chance of producing an offspring that is more fit, regardless of the parents' fitness. However, by itself free recombination causes a progressive loss of information: those genes that are amplified replace others that are lost forever. This loss of diversity in the gene pool is disastrous if the ensemble of sources changes, as demonstrated in Experiment 2 (see Appendix A). The mutation operator continuously injects diversity into the gene pool, thereby preventing the system from becoming trapped in a low-diversity dead end.

Our approach differs from some genetic-algorithm and neural-network approaches in a fundamental way. We do not seek an individual network that is "most fit" overall; instead, we seek subpopulations of networks that have specialized their responses to particular sources. The response of the system is an aggregate, macroscopic feature of the individual responses of a large population of individual, interacting subsystems. We view fitness as a very general concept: simply a measure of the similarity between the input and the output. Rather than being built in to the fitness function, the evolutionary trend toward specialization is instead an emergent property of the population as a whole, and a consequence to the informational bottleneck in the encoders. Unlike the more standard optimization methods for designing systems, this method results in subpopulations that resemble species adapted to different ecological niches that are determined by the sources.

1.3.2 Recombination: A Genetic Operation

Our second domain of study was the evolution of the recombination as a genetic operation. The evolution of a selectively neutral modifier of recombination is studied under different conditions of selection on the major genes. In a finite population a simulation study is carried out in which the phenotype is computed additively from the genotype at twenty genes. The fitness is taken to be a function of the phenotype and we show that when this function is very jagged, low recombination has a strong advantage. When the function is smooth and of the disruptive-selection kind, high recombination may be favored in both finite and very large populations. In a deterministic numerical study of disruptive selection on two loci it is shown that the evolution of recombination depends on

the initial frequencies at the selected loci, on the exact shape of selection and on the strength of the selection. In general, when the selection is disruptive and very strong, it is possible to find conditions under which higher recombination will be favored.

We found a delicate dependencies on the shape and strength of the disruptive selection, on the initial average phenotype and its distribution, and on the distribution of high recombination allele, CH, among the selected chromosomes which conspire to make generalizations very difficult. Perhaps the only general conclusion we may draw is that when disruptive selection is strong, there will be a set of initial chromosome frequency vectors in the population from which evolution will favor CH. On the other hand, under the same conditions CL will usually be favored for some other set of starting conditions. As selection becomes stronger, the latter set appears to decrease in size relative to the former.

1.4 Future Work

In the future we would like to investigate three research areas: two processes described in the previous section and the third described in the body of the report. First we would like to attain a better and formal understanding of the relation between the feature maps generated by Kohonen's network and the generalization of the system we have been investigating. A detailed outline of the approach will be discussed later. This work will be tightly linked to the investigation of dimensionality reduction, where the dimensions under consideration are the geometrical organization of the individuals in the population.

The second area of research will be on the evolution of learning capabilities. This research will lead to a better undesrstanding of the conditions under which learning mechanism as opposed to fix algorithm is advantagous. It will reflect also on the question of what should be the number of learning steps before performing a genetic operation like recombination and mutation.

The third proposed direction is the investigation of the effect of coevolutionary processes on the formation of clusters in the population and maintaining variability in a controlled way to preserve memory of past experience in the presence of a changing environment.

The results of the research will lead to better understanding of the relationship among neural network theory, evolutionary and population genetics, and some aspects of dynamical systems theory. We expect also that fields such as signal processing and machine learning will greatly benefit from the outcome of this research.

2 Appendix A

2.1 Definition of the Problem

Suppose that we have a system, for the time being regarded as a "black box," that receives as input a *signal vector* of length n, $\mathbf{x} = \langle x_0, \ldots x_{n-1} \rangle$. These signals could be, for example, speech waveforms. The components of $\mathbf{x}$ are real numbers within some limited dynamic range. In practice, since any measurement of a real signal will be uncertain to some degree, we can represent the signal vector with nonnegative integers to some precision b bits. Each possible signal is a point in the n-dimensional metric *signal space*.

Now suppose the system is stimulated only by a much smaller, structured ensemble of signals generated by a few unknown, relatively low-dimensional physical processes, possibly corrupted by noise. They are called *sources*. They could be, for example, a few speakers of English. There may be considerable variation within a single source, so we should imagine a source to be represented by a subset of the signal space: its *attractor*. The task of the system is to respond distinctively to each source. From looking at a macroscopic feature of the system, we should be able to tell when it has been presented with a source and which source it is.

In the simplified problem we restrict the components of the input vector to binary values ($b = 1$) and restrict the sources to single values (point attractors). Under these assumptions, the system will be learning a subset of the numbers $\{0, \ldots, 2^n - 1\}$. The signal vector can be visualized as the corners of an n-dimensional hypercube, and the response of the system will be to select one of these corners.

2.2 Encoder Populations

Each subsystem is an instantiation of a simple neural network called an *encoder* [1,11] as shown in Figure 1. An n_1-n_2-n_3 encoder has n_1 inputs that feed into n_2 hidden units, which in turn feed into n_3 output units. Each unit computes a weighted sum of the inputs and compares the result with a threshold. If the sum exceeds the threshold, the unit is activated and outputs a one; otherwise, it produces a zero.

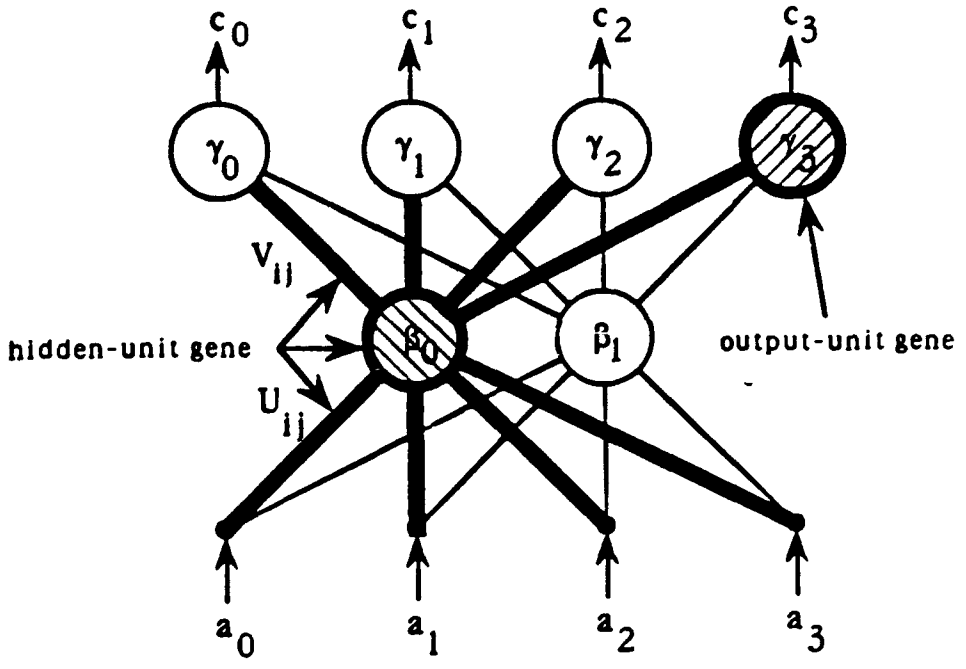

Figure 1: A 4-2-4 encoder.

Originally, these networks were used to attack the *encoding problem* [11]. Assume that $n_1 = n_3$ and $n_2 = \log_2 n_1$, and that the inputs consist of a single one bit, with all the rest zeros. The position of this bit then represents one of the first n natural numbers. The encoding problem is to learn to encode these numbers into a pattern of $\log n$ bits, and also to learn to decode this $\log n$ bits pattern into an output pattern, usually identical to the input pattern. We, however, are using the population of encoders in quite a different way. Instead of finding a single network that solves the encoding problem for all sources, we want to construct subpopulations of networks that are specialized for encoding different sources.

In general, an encoder is a tuple

$$\xi = [\beta, \gamma, U, V]$$

where $\beta = \langle \beta_0, \ldots, \beta_{n_2-1} \rangle$ and $\gamma = \langle \gamma_0, \ldots, \gamma_{n_3-1} \rangle$ are thresholds for the hidden units and the output units, respectively, and $U = \{u_{ij} | 0 \leq i < n_2, 0 \leq j < n_1\}$ and $V = \{v_{ij} | 0 \leq i < n_2, 0 \leq j < n_3\}$ are weight matrices.

An encoder accepts an n_1-bit input vector $\mathbf{a}$, produces an n_2-bit hidden vector $\mathbf{b}$, and then produces an n_3-bit output vector $\mathbf{c}$. Each unit applies a threshold function

$$\Theta(\phi, s) = \begin{cases} 1 \text{ if } s \geq \phi \\ 0 \text{ otherwise} \end{cases}$$

to the sum of its weighted inputs:

$$b_i = \Theta(\beta_i, \sum_{0 \leq j < n_1} u_{ij} a_j)$$

$$c_j = \Theta(\gamma_j, \sum_{0 \leq i < n_3} v_{ij} b_j) \ .$$

It is essential to the genetic algorithm described below that a description of an encoder may be decomposed into parts, called *genes*, in such a way that a new encoder (a child) can be cor ructed with parts from two others (the parents) [7,5]. In part, we have chosen the encoder network for this work because it can be decomposed in a fairly natural way. The genetic structure of an encoder is illustrated in Figure 1. Each encoder has n_2 hidden-unit genes and n_3 output-unit genes. The hidden-unit genes are the more complex of the two types. The ith hidden-unit gene of an encoder ξ consists of the hidden-unit threshold β_i, a vector of input weights $\langle u_{ij} | 0 \leq j < n_1 \rangle$, and a vector of hidden-unit weights $\langle v_{ij} | 0 \leq j < n_3 \rangle$. The jth output-unit gene consists simply of the output-unit threshold γ_j.

The system consists of a population of N encoders

$$\Xi = \{\xi_k, 0 \leq k < N\}$$

with, in general, different thresholds and weights. We always have $n_1 = n_3$ and typically, but not necessarily, $n_2 = \log_2 n_1$. Every encoder in the population is presented simultaneously with the same input vector, and tries to reconstruct the input. Success is measured by a *fitness function* [3,8]

$$f_k(\mathbf{a}) = - \sum_{0 \leq j < n} |a_i - c_i| \ .$$

Note tha' fitness is simply the negative of the Hamming distance between the input and the output vectors. The idea behind the genetic algorithm described below is to increase the frequency of genes and combinations of genes in Ξ by selection, thereby causing the population to learn to encode the inputs it sees most frequently.

2.3 A Genetic Algorithm

Genetic algorithms can be effective for exploring large design spaces [5,7]. The essential idea is to simulate many generations of populations of individual subsystems, with each generation produced from previous generations by selection and differential reproduction [3,4,6,10]. Each individual is graded by a fitness function that is intended to measure its performance on one or more instances of a problem. Those individuals that are most fit are selected and then a set of new subsystems is created by applying genetic operators to the descriptions of the selected individuals. Commonly used genetic operators are called *crossover* and *mutation*, modeled after similar processes that drive biological evolution [2,5,7]. Although the concepts behind genetic algorithms are very general, there are inevitably a wide variety of parameters, reproduction schemes, representations, and so on that could be used. Part of the aim of this preliminary work is to understand the consequences of and interactions among these choices.

Our genetic algorithm consists of an initialization,

$$\Xi \leftarrow \Xi^0$$

followed by an iteration of the generation operator, $\mathcal{G}$:

$$\Xi \leftarrow \mathcal{G}(\Xi, \mathbf{a}^t),\ t = 0, 1, \ldots$$

In the initialization step, a population of at least $N = 4096$[1] encoders with n inputs and m hidden units is created. All thresholds and weights are chosen from a uniform random distribution over the interval $[-1, 1]$. Initially, all of the members of Ξ are marked as alive and are assigned an age chosen from a random distribution of integers in the range $[0, \ldots, age_{max} - 1]$. Only those encoders marked as alive, denoted by Ξ_a, are active and available for input, selection, and reproduction. All encoders that are not alive are treated as available space for the next generation. The age of ξ is an integer indicating the number of generations for which ξ has been continuously alive.

[1] We use a Connection Machine with 4096 processors for our simulations. N can be larger than 4096, but must be a power of 2.

The generation function $\mathcal{G}$ is defined as the following sequence of steps:

$$\begin{aligned}
\Omega &\leftarrow \mathit{select}(f, \Xi, \mathbf{a}) \\
\Omega^* &\leftarrow \mathit{reproduce}(\Omega) \\
\Xi &\leftarrow \mathit{insert}(\Omega^*, \Xi) \\
\Xi &\leftarrow \mathit{age}(\Xi) \\
\Xi &\leftarrow \mathit{kill}(\Xi)
\end{aligned}$$

These steps can be performed in several ways, but each step has the basic characteristics outlined below, in Section 2.4.

Selection: $\Omega \leftarrow \mathit{select}(f, \Xi, \mathbf{a})$

An input bit vector, $\mathbf{a}$, is chosen and presented to the system. The input can be selected in a variety of ways. The simplest is to select the vector from a set of sources according to some prior probability distribution. Input vectors can be degraded with noise by inverting bits with some probability. Inputs can also be chosen randomly from the set of 2^n possible inputs with some specified frequency. All living encoders are ranked by fitness and a subset Ω of the most fit is selected. The size of Ω could be determined dynamically by a threshold on fitness. Instead, in this preliminary investigation, we set the size of Ω as a fixed proportion of the size of Ξ (usually 1/16).

Reproduction: $\Omega^* \leftarrow \mathit{reproduce}(\Omega)$

Every member of Ω is paired at random with another member of Ω (possibly itself), which is called its mate. The pairs are combined to produce a fixed number of children. The combination is performed by applying two genetic operators, crossover and mutation. In the crossover operation, every child's gene is selected from one or the other parent with probability 1/2, a process called *free recombination* [6,9]. In the mutation operation, every gene constituent, whether a weight or a threshold, is replaced by a random value with some probability of mutation μ, which is usually quite low.

Insertion: $\Xi \leftarrow \mathit{insert}(\Omega^*, \Xi)$

A random number $k \in \{0, \ldots, N-1\}$ is generated for every child in Ω^*. If ξ_k is not alive, the child is inserted into Ξ at that location, is marked as alive, and is assigned an age of zero. If more than one child tries to occupy the same location, one child is chosen at random.

Aging: $\Xi \leftarrow \mathit{age}(\Xi)$

The ages of all living encoders are increased by 1.

Death: $\Xi \leftarrow kill(\Xi)$

Every encoder whose age is greater than age_{max} is marked as not alive. Its space in Ξ then becomes available for the children in the next generation.

2.4 Results

When interpreting the performance of the system, we consider only those encoders that can reconstruct their outputs perfectly. These are said to respond to the input; that is, $r_k(\mathbf{a}) = 1$, where

$$r_k(\mathbf{a}) = \max(0, 1 + f_k(\mathbf{a})) \ .$$

We want many networks to respond to the sources, few or none to respond to nonsource signals, and different subpopulations to respond to each different source.

Two measures of the effectiveness of the system depend on computing the probability distribution $P(\mathbf{a}|r)$, which is the probability that the signal is $\mathbf{a}$ given that a randomly chosen encoder is responding. This distribution is computed assuming no prior knowledge of the frequency of occurrence of the source. Therefore, using a uniform (maximum entropy) distribution of priors

$$P(\mathbf{a}) = \frac{1}{2^n}$$

and writing the probability of an encoder responding to $\mathbf{a}$ as

$$P(r|\mathbf{a}) = \frac{\sum_k r_k(\mathbf{a})}{N} \ ,$$

and the probability of an encoder responding to any signal as

$$P(r) = \frac{\sum_{\mathbf{x}} \sum_k r_k(\mathbf{x})}{N 2^n} \ ,$$

we use Bayes's Rule to determine the desired distribution:

$$P(\mathbf{a}|r) = \frac{P(r|\mathbf{a})P(\mathbf{a})}{P(r)} \ ,$$

or

$$P(\mathbf{a}|r) = \frac{N \sum_k r_k(\mathbf{a})}{\sum_{\mathbf{x}} \sum_k r_k(\mathbf{x})} \ .$$

Ideally, this distribution should be identical to the prior probability $P(\mathbf{a})$ after many generations.

We can compute the entropy of $P(\mathbf{a}|r)$

$$S = -\sum_{\mathbf{x}} P(\mathbf{x}|r) \log_2 P(\mathbf{x}|r)$$

to summarize the degree of organization of the system in terms of the uncertainty associated with its response. We can also compute the correlation between $P(\mathbf{a}|r)$ and some prior model distribution $P_M(\mathbf{a})$ from which the sources were chosen:

$$C = \frac{\sum_{\mathbf{x}}(P(\mathbf{x}|r) - \overline{P(\mathbf{x}|r)})(P_M(\mathbf{x}) - \overline{P_M(\mathbf{x})})}{\sqrt{\sum_{\mathbf{x}}(P(\mathbf{x}|r) - \overline{P(\mathbf{x}|r)})^2}\sqrt{\sum_{\mathbf{x}}(P_M(\mathbf{x}) - \overline{P_M(\mathbf{x})})^2}} .$$

The first three experiments described below use entropy and correlation to examine the evolution of the system under different conditions. Because the time required to compute $P(\mathbf{a}|r)$ grows exponentially with the length of the input vector, n, these experiments were done only on small 4-2-4 encoders. The fourth experiment examines the behavior of the system when n is larger and, in particular, when the number of possible inputs greatly exceed the size of the population. Finally, the fifth experiment examines whether the population becomes specialized to the sources.

2.4.1 Experiment 1: Typical Behavior (no mutation)

The first experiment examines the typical behavior of a population of 16K 4-2-4 encoders with no mutation ($\mu = 0$). The inputs were chosen at random with equal frequency from a set of four sources. Figure 2 shows the entropy of $P(\mathbf{a}|r)$ over 1000 generations when the maximum number of children n_c is 2 and 4 ((a) and (b), respectively). Also shown is the size of the population that is living.

In both cases the entropy eventually drops to the ideal value of $\log_2 4 = 2$, which is the entropy of the model distribution. The correlation with the model distribution (not shown) is very nearly 1 after only about 20 generation. The fraction of the population that is living fluctuates at first, but eventually approaches some limit, which is greater for the $n_c = 4$ case.

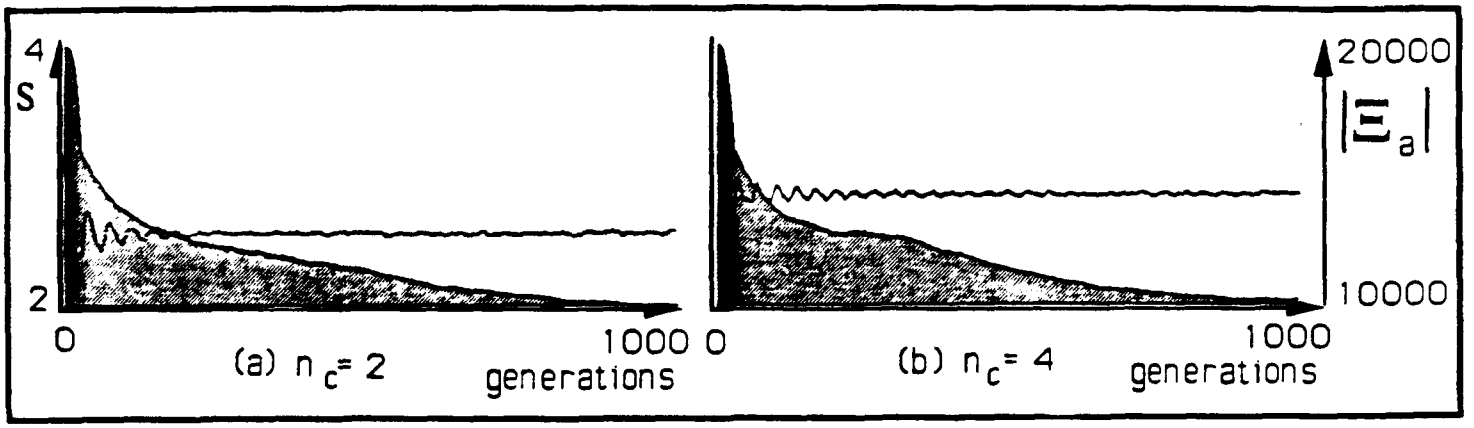

Figure 2: Typical behavior (no mutation)

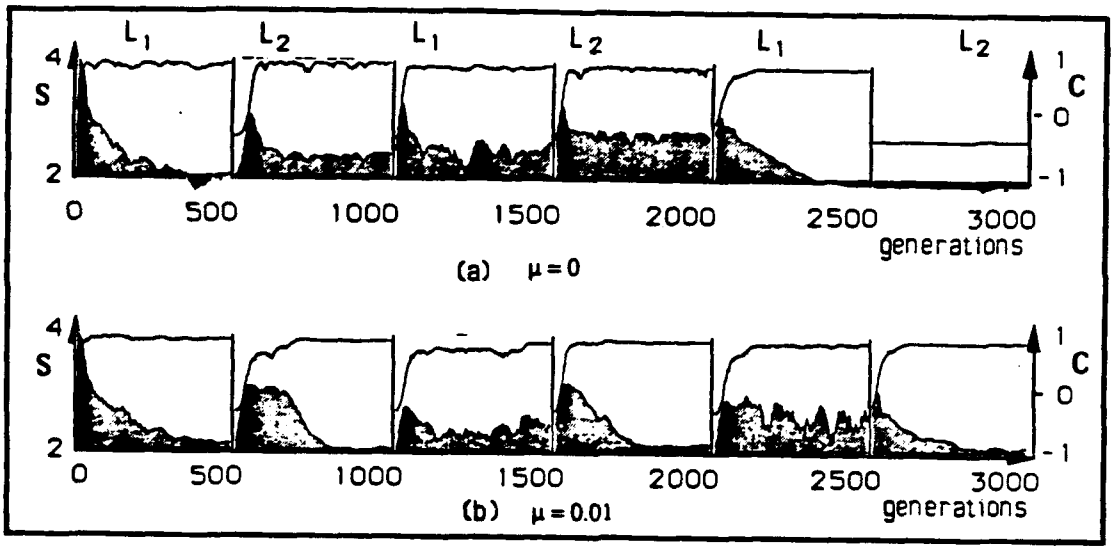

Figure 3. Changing environment

2.4.2 Experiment 2: Changing Environment

The previous simple experiment illustrated that adaptation can occur without mutation, relying only on the crossover operation. This experiment shows that mutation is essential in a more challenging problem. Figure 3 shows the entropy and the correlation measures when the system is successively stimulated with two different sets of four signals, L_1 and L_2. Two cases are shown: $\mu = 0$ and $\mu = 0.01$. The interesting feature of this experiment is that in the first case, $\mu = 0$, the system

"collapses" into an irreversible condition of total insensitivity on the third presentation of the set L_1. The entropy drops to zero, indicating that the system can respond to no signals (or possibly to only one), and the correlation with the model distribution drops effectively to zero. Apparently, the successive presentations and epochs of selection have eliminated variation in Ξ. Selection for L_1 eliminates genes effective for L_2, selection for L_2 eliminates genes effective for L_1, and so on, until by the third presentation of L_1, Ξ has been so depleted that it cannot adapt.

In the case of $\mu = 0.01$ this does not happen. Even this low rate of mutation is sufficient to maintain adequate variation in Ξ. The crossover operation is effective for making large jumps though the space of genotypes, while mutation is effective as a continual source of variation.

2.4.3 Experiment 3: Effects of Noise

Experiment 3 examines the effects of noise in the input. The population size is 4K, the encoders are 4-2-4, four different sources are used with equal probability, $\mu = 0.01$, $n_c = 4$, and $age_{max} = 30$. Each encoder is presented with an input vector, selected from the four sources, but each vector has a probability P_n of having (at least) one bit changed at random. All encoders receive input from the same source, but the inputs are corrupted by noise independently, so that any two encoders may see different signals. Figure 4 shows four cases: $P_n = 0.1, 0.2, 0.25, 0.4$. Entropy is shown above and correlation below. The shaded portions of the correlation graphs indicate when the system is working, in the sense that the four signals of highest probability are identical to the sources. The system performs well up to $P_n = 0.2$ but degrades quickly for higher noise levels.

2.4.4 Experiment 4: Large n

To test the system on a larger problem, and in particular on a problem in which the number of possible signals greatly exceeds the size of Ξ, we performed a simulation with 16-4-16 encoders and eight sources. As in the previous simulation the population size is 4K, $\mu = 0.01$, $n_c = 4$, and $age_{max} = 30$. Because the number of possible inputs is $2^{16} = 64K$ it is not practical to compute the complete distribution $P(\mathbf{a}|r)$, especially not for every generation. Instead, we let the system run for 4,000 generations and then counted the number of encoders that responded averaged over all eight

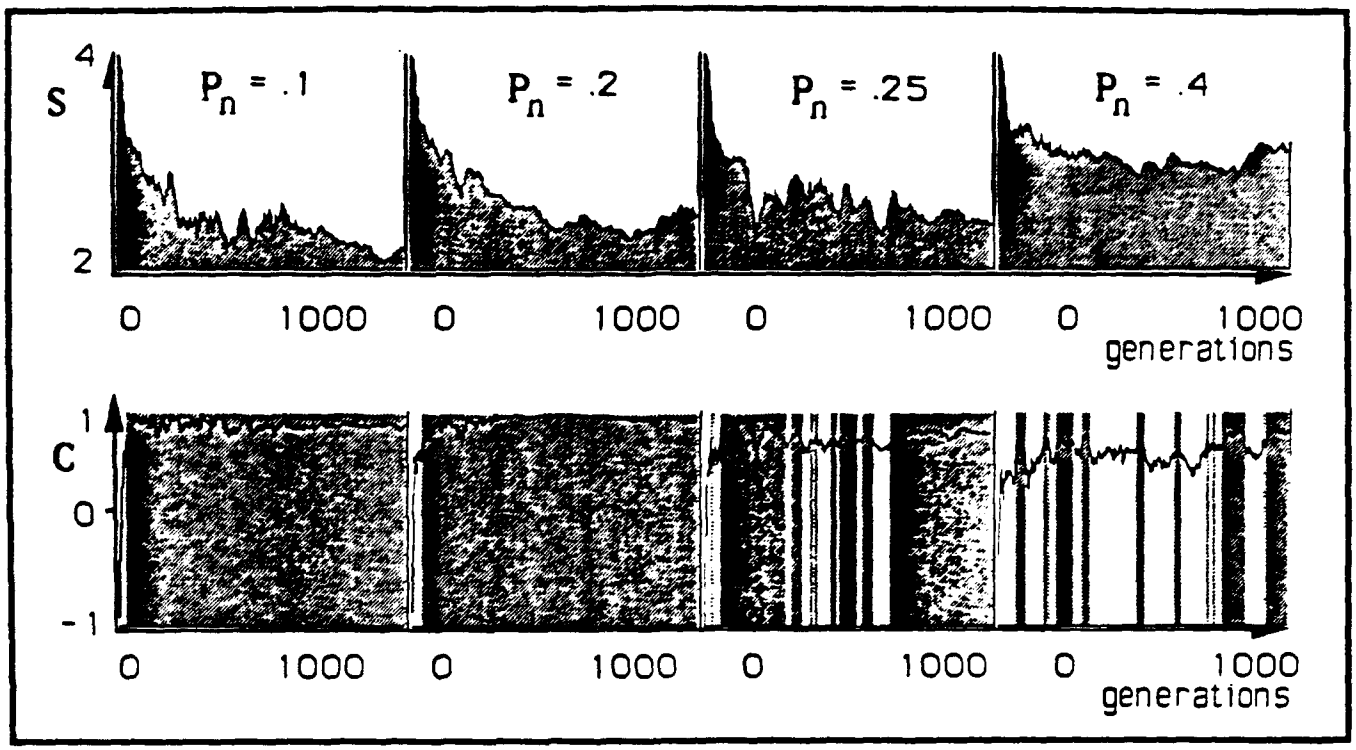

Figure 4: Effects of noise

sources, which was 488.5, and the average number of encoders that responded averaged over 1000 randomly chosen signals, which was 0.13.

2.4.5 Experiment 5: Specialization

The last experiment examines whether the population divides into disjoint subpopulations specialized for the sources. Suppose we have s sources with R_i being the subpopulation of encoders that respond to source i. The following equation gives a normalized measure of the overlap between two subpopulations:

$$O_{ij} = \frac{|R_i \cap R_j|}{|R_i \cup R_j|} \quad 0 \leq i, j < s \; .$$

Ideally, O_{ii} should be one if $i = j$ and zero otherwize for complete specialization. Figure 5 shows matrices of overlap measures for four cases. When we adapt 4-2-4 encoders to only two sources, shown in Figure 5 (a), no specialization occurs at all: nearly every encoder that responds to one source also responds to the other. When we adapt the same system to four sources (b) or seven sources (c), there is some specialization, with relatively more specialization occurring when there are more sources.

2 sources overlap matrix (for 4-2-4 encoders)

	S1	S2
S1	1.0	0.99
S2	0.99	1.0

(a)

4 sources overlap matrix (for 4-2-4 encoders)

	S1	S2	S3	S4
S1	1.0	0.99	0.99	0.0
S2	0.99	1.0	0.99	0.0
S3	0.99	0.99	1.0	0.0
S4	0.0	0.0	0.0	1.0

(b)

7 sources overlap matrix (for 4-2-4 encoders)

	S1	S2	S3	S4	S5	S6	S7
S1	1.0	0.22	0.62	0.53	0.0	0.0	0.0
S2	0.22	1.0	0.43	0.33	0.0	0.0	0.0
S3	0.62	0.43	1.0	0.69	0.0	0.0	0.0
S4	0.53	0.33	0.69	1.0	0.0	0.0	0.0
S5	0.0	0.0	0.0	0.0	1.0	0.99	0.99
S6	0.0	0.0	0.0	0.0	0.99	1.0	0.99
S7	0.0	0.0	0.0	0.0	0.99	0.9	1.0

(c)

10 sources overlap matrix (for 16-2-16 encoders)

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S2	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S3	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
S4	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.66
S5	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
S6	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
S7	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
S8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
S9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
S10	0.0	0.0	0.0	0.66	0.0	0.0	0.0	0.0	0.0	1.0

(d)

Figure 5: Specialization

Finally, when we adapt a system of 16-2-16 encoders to ten sources, Figure 5 (d), the specialization is nearly perfect, with only two subpopulations having a significant degree of overlap.

2.5 Encoder Populations: Conclusions

For the encoding problem, the evolutionary algorithm exhibits effective adaptation. Differential reproduction amplifies the frequency of selected genes and leads to the emergence of a population that is progressively more fit. In our model, free recombination (crossover) seems to be the primary means of adaptation. Two relatively fit parents clearly have a better-than-average chance of producing more fit offspring. Mutation, on the other hand, has only an average chance of producing an offspring that is more fit, regardless of the parents' fitness. However, by itself free recombination causes a progressive loss of information: those genes that are amplified replace others that are lost forever. This loss of diversity in the gene pool is disastrous if the ensemble of sources changes, as demonstrated in Experiment 2. The mutation operator continuously injects diversity into the gene pool, thereby preventing the system from becoming trapped in a low-diversity dead end.

Our approach differs from some genetic-algorithm and neural-network approaches in a fundamental way. We do not seek an individual encoder that is "most fit" overall; instead, we seek subpopulations of networks that have specialized their responses to particular sources. The response of the system is an aggregate, macroscopic feature of the individual responses of a large population of individual, interacting subsystems. We view fitness as a very general concept: simply a measure of the similarity between the input and the output. Rather than being built in to the fitness function, the evolutionary trend toward specialization is instead an emergent property of the population as a whole, and a consequence to the informational bottleneck in the encoders. Unlike the more standard optimization methods for designing systems, this method results in subpopulations that resemble species adapted to different ecological niches that are determined by the sources.

We would like to simulate populations with more diverse features, such as variable sizes, reproduction rates, age limits, and mutation rates. Currently, these properties are global to all encoders, but they could be variable, inherited properties, represented as "modifier genes" attached to the basic encoder genotype. We speculate that this process will lead to more interesting adaptation because it

will create more niches for adaptation to fill. For example, one can imagine relatively large, scarce, long-lived encoders specializing on complex sources that appear infrequently or change slowly or relatively small, numerous, short-lived, and perhaps highly mutable encoders specializing on common, simple sources.

We are changing the input representation of the more general case of b-bit samples so that we can investigate applications to real, physical sources. Whether the approach can be extended to more complex sources than point attractors is an open question. To do so, the basic encoder representation may have to be extended to a more elaborate, dynamic network. Instead of an encoder, we may need a generator whose internal state allows it to recognize and mimic (i.e., predict) a sours with a low number of dimensions.

Variability is one of the important driving forces that causes a population to evolve. One way of maintaining variability in population is by mutation, but mutation is a random process that causes a reduction in the population performance and may lead, together with drift, to an unfit population. Are there more sophisticated mechanisms by which nature chooses to operate? Is coevolution a process that can be artificially reproduced and generate populations that will be able to adapt to a changing environment while memorizing the important features of the history? Our experiments indicate that such a mechanism and behavior can be mimicked and rather interesting dynamical behavior can be observed. The question that can be asked here is the relation of such systems to dissipative dynamical systems, where the environment acts as an energy source and the parasites ("viruses") act as the dissipative part of the system.

2.6 Coevolution

Thus far in our research we have dealt with individual populations in isolation having no interaction with other populations. However, in the natural world populations do not exist in isolation. The interactions (between populations) are intrinsically interesting because they produce perhaps the most intricate and fascinating patterns in biology. In this section we will introduce the notion and implication of the evolution of population of processes in the presence of "parasites." A parasite can be considered a low-level process which depends on its host for survival and reproduction. The

host provides the environment for the parasite, and as long as the parasite can exploit the host, it can survive while causing harm to its host (a synergetic behavior can also be included, but for the arguments below I will not consider that option). Consider the following interaction between a host and its environment and a parasite and its environment, i.e. the host itself. The host is selected such that its fitness is maximized. The parasite reproduces, and is therefore considered successful, if it can exploit its host, say, by recognizing its genetic makeup. Once recognition is achieved, the host is no longer operational and the parasite spreads its offspring, copies of itself, to neighboring hosts. In the case where the neighboring hosts have similar genetic makeup to that of the original host, in the next generation they will be non-functional and will no longer produce offspring. The hosts that survive are those that have enough variability in their genetic makeup to avoid the parasite. Since the host is subjected to its environment and the process of selection causes the elimination of the processes that responds poorly to the environment, the processes that survive are the processes that are successful in responding to the environment and simultanously avoiding the parasite. Such a behavior could be achieved if the variation in the host is such that it occurs in places that are not critical for the selection process that occur at the phenotypic level. For example, consider a process which is the conversion of a binary bit string to its integer representation. The parasite, a binary bit string of the same length, looks only at the binary bit string and measures its Hamming distance to it regardless of its integer representation. In case the selection is based on the highest integer representation for the host, the variability that will be maintained in the host that will have the minimal effect on its phenotypic fitness and still maintain high distance from the parasite will be at the least significant bits. Such a behavior of controlled variability is better than random mutation since its effect on the phenotypic level is minimal while random mutation has no bias to maintain high fitness at the phenotypic level.

We have shown (in preparation) that, in the presence of changing environment, the coevolved population in the presence of a parasite can evolve to fit the new environment while maintaining a memory about the past environment, longer than when variation is maintained by random mutation.

3 Appendix B

3.1 Recombination Dynamics

In this section we will concentrate on a study we did on the conditions under which recombination in a genetic systems is favored. This study includes analytical and numerical results.

The evolution of a selectively neutral modifier of recombination is studied under different conditions of selection on the major genes. In a finite population a simulation study is carried out in which the phenotype is computed additively from the genotype at 20 genes. The fitness is taken to be a function of the phenotype, and we show that when this function is very jagged, low recombination has a strong advantage. When the function is smooth and of the disruptive-selection kind, high recombination may be favored in both finite and very large populations. In a deterministic numerical study of disruptive selection on two loci it is shown that the evolution of recombination depends on the initial frequencies at the selected loci, on the exact shape of selection, and on the strength of the selection. In general, when the selection is disruptive and very strong, it is possible to find conditions under which higher recombination will be favored.

Recent research on the evolution of recombination has demonstrated that a selectively neutral genetic modifier of recombination, introduced near an equilibrium of a large randomly mating population that is in linkage disequilibrium, will succeed if it reduces recombination (Feldman et al., 1980; Feldman and Liberman, 1986; Liberman and Feldman, 1986). Similar reduction results hold for modifiers of mutation and migration (Liberman and Feldman, 1986, 1989). There are a series of mathematical and numerical caveats to this Reduction Principle that involve departures from the assumption of the modelling framework under which the principle was derived. Among these are the following.

Nonrandom mating: Numerical studies (Charlesworth et al., 1979) and analytical work (Holsinger and Feldman, 1983) have shown that, in the presence of inbreeding, a modifying allele that increases the value of the parameter under its control may succeed.

More general forms of constant selection: The Reduction Principle was proved under con-

stant viability selection. When selection operates at the level of fertility (Holsinger et al., 1986) or with different viabilities for males and females (Twomey and Feldman, 1990), the Reduction Principle may fail. When it fails, however, it usually does so only when the linkage between the selected genes and the modifier is sufficiently loose. Similar results are true when segregation distortion occurs at the major loci (Thomson and Feldman, 1974; Feldman and Otto, 1990).

Cyclically fluctuating selection: Charlesworth (1976) showed numerically that if selection favors the coupling and repulsion phases in an alternating way with a period of two or more generations, increase of recombination may occur.

In another series of studies with recombination modifiers two additional features were introduced: **finite population size** and **different starting conditions**. Felsenstein and Yokoyama (1976) studied a finite population in which fitnesses were multiplicative and directional and the population was initially fixed at each locus. Variability was introduced by mutation to favorable alleles. Under these assumptions high recombination tended to be favored, a result which depended on the mutation rate but not on the strength of selection. Maynard Smith (1979, 1980, 1988) took a Gaussian distribution for the phenotypic values and studied the evolution at a recombination controlling locus at which the high and low allele were initially equally frequent, that is, not in the neighborhood of an equilibrium as is required for the Reduction Principle. Some of these models have been reexamined by Bergman and Feldman (1990) with the following conclusions. When the phenotypic and selective optima coincide, recombination is reduced. When the selective optimum is shifted, the results depend on the strength of the selection, i.e. the variance of the Gaussian regime. Under strong selection, reduction occurs, while if the selective mean is shifted far enough from the phenotypic mean, higher recombination may evolve. There is critical dependence here on the variance of the Gaussian selection regime, even in finite populations. In the latter case with small variance (i.e. strong selection) lower recombination was shown to be strongly favored while the results are equivocal with weaker selection. When the mean of the selection distribution is shifted far enough away from the phenotypic mean, higher recombination is favored in the finite population analyses, although once again there is a marked effect of the variance. In these analyses under Gaussian selection, the sign of the linkage disequilibrium is not diagnostic of whether high or low recombination succeeds (Bergman and Feldman, 1990).

It should also be recalled that Maynard Smith (1979) found in a disruptive selection model with different selective optima in different niches, and Levene-type population structure (Levene, 1953), low recombination was favored.

The array of these recent results led us to propose that it would be useful to phrase the question of the evolution of recombination in terms of the structure of the selection function in more general terms. In the present study we first report on results for a 20-locus diploid model, with an additional recombination-controlling locus, in a finite population model. The fitness functions used are defined in terms of different numbers of coefficients in a harmonic series. The results of these numerical simulations suggested that certain forms of selection functions might be the most conducive to the evolution of high recombination. These regimes, characteristic of some views of disruptive selection, were investigated in more detail.

3.2 Finite Population Model

One hundred diploid individuals each defined at 20 loci are considered. There are two alleles at each of these 20 loci which are under selection. An additional twenty-first gene controls the amount of recombination across the whole chromosome. At each locus the alleles are labelled 0 and 1 and the phenotype of an individual is constructed by summing the 1's at the 20 selected loci. Thus, the phenotypic value, ν takes on values between 0 and 40.

Since the population size is small relative to the 2^{20} chromosome types possible at the selected loci, we choose the initial population according to some probabilistic rule. For the results reported here a 1 or a 0 were assigned equally likely at each allele of each locus, and this procedure was done independently between loci. On average this gives an initial mean phenotype of 20 and linkage equilibrium.

From the 100 individuals two parents are chosen at random and each donates a gamete, after recombination, to an offspring. The offspring's phenotype is then evaluated against the fitness function and, if it survives, it is listed as part of the next generation. This process is repeated until 100 offspring have survived.

The Fitness Landscape. The phenotypic value ν takes values between 0 and 40. Set $x = \nu/40$ and define the function

$$\psi(x) = \sum_{k=1}^{n} \frac{R(k) \sin \pi k x}{\sqrt{k}} \tag{1}$$

where $R(k)$, $k = 1, 2, \ldots, n$ are random numbers uniformly distributed on $[0, 1]$. We then define the fitness of the phenotype with values ν as

$$F(\nu) = \frac{\psi(x) - \min[\psi(x)]}{\max[\psi(x)] - \min[\psi(x)]}. \tag{2}$$

The "jaggedness" or, as Kaufman and Levin (1990) called it, "ruggedness" of the fitness function is controlled by varying n; the greater is n, the more jagged is $F(\cdot)$. We have examined in detail $n = 2, 3, 20, 40$. After n is fixed, $R(k)$, $k = 1, 2, \ldots, n$ are chosen. The initial configuration of the population is chosen and the simulation proceeds until one or other allele at the recombination controlling locus is fixed. For each set $R(k)$ the simulation is repeated 500 times each with a randomly chosen starting population.

For each value of n, 50 different sets of $\{R(k)\}$ are chosen and the results of the 50 × 500 runs for each n constitute the data. A control experiment where the twenty-first locus had no effect on recombination was also carried out.

Recombination is controlled by the (neutral) twenty-first locus with alleles we call CL and CH. Genotype CL/CL produces a probability 0.01 that there is at least one break per pair of chromosomes while for CH/CH this probability is 0.50. In the dominant case CH/CL also produces 0.50 and in the recessive case CH/CL gives a recombination rate of 0.01. If a break occurs its position is chosen uniformly across the 21 genes. Up to three breaks are permitted, with 1, 2, or 3 breaks being equally likely, given that recombination occurs, according to the above probabilities. In choosing the breaks uniformly across the loci, no single position was permitted to be chosen twice. Following recombination but before selection, one of three kinds of mutation regimes was imposed. In the first, there was no mutation; in the second, there was symmetric mutation, i.e. from 0 to 1 and 1 to 0 at rate 0.005 per locus; and in the third, mutation was unidirectional, from 0 to 1 only at this rate.

The initial frequency of the high recombination allele CH was 5 percent in the population. The simulation was pursued until either CH or CL was fixed. For each set of 500 runs with a given choice of

$R(k)$ the number which resulted in fixation of CH was tabulated and the distribution of these numbers among the 50 different sets of $R(k)$ that were chosen for each n is recorded in Table 1 (dominant case) and Table 2 (recessive case).

3.3 Results of Finite Population Model

Tables 1 and 2 record the distributions of observed frequencies of fixation of CH according to the jaggedness of the fitness landscape F as specified by the number of coefficients n. If CH were completely neutral in its effect on the whole genotype, we would expect 5 percent, or 25, of the 500 runs to fix on CH. The tables record the results in histogram form as a function of $n = 2, 3, 20, 40$. The most obvious feature of the tables is that with 20 or 40 coefficients, low recombination is favored. If anything, this advantage is stronger in the recessive case (see also Bergman and Feldman, 1990). On the other hand, there is some advantage to high recombination in the cases $n = 2$ or 3, but it is not nearly as strong an effect as the advantage of low recombination with $n = 20$ or 40. The role of mutation does not appear to be qualitatively important in the dominant case although the presence of mutation in the recessive model does seem to enhance the effect in favor of CL.

Table 1. High Recombination Dominant.

Distribution of fixations of the high recombination allele.

Model	Number out of 500 with high-recombination allele fixed							
Mutation*	≤ 10	11-15	16-20	21-25	26-30	31-35	36-40	≥ 41
from 0 to 1								
2 coefficients	0	0	4	14	12	12	5	3
3 coefficients	0	1	8	14	12	6	4	5
20 coefficients	18	19	7	5	1	0	0	0
40 coefficients	24	16	5	4	1	0	0	0
Symmetric								
mutation**								
2 coefficients	0	1	9	11	19	7	1	2
3 coefficients	0	4	6	15	14	8	1	2
20 coefficients	29	9	5	6	1	0	0	0
40 coefficients	26	12	10	0	2	0	0	0
No Mutation								
2 coefficients	0	0	6	13	19	5	3	4
3 coefficients	0	1	6	21	11	8	2	1
20 coefficients	29	10	10	1	0	0	0	0
40 coefficients	36	9	1	4	0	0	0	0

*Mutation is at 0.005 per locus per generation.

**Mutation from 0 to 1 and 1 to 0 each at 0.005 per locus per generation.

Table 2. High Recombination Recessive.

Distribution of fixations of the high recombination allele.

Model	Number out of 500 with high-recombination allele fixed							
Mutation*	≤ 10	11-15	16-20	21-25	26-30	31-35	36-40	≥ 41
from 0 to 1								
2 coefficients	1	7	12	10	6	7	7	0
3 coefficients	0	1	3	5	8	14	10	9
20 coefficients	10	8	12	6	10	4	0	0
40 coefficients	47	0	2	1	0	0	0	0
Symmetric								
mutation**								
2 coefficients	1	7	12	10	6	7	7	0
3 coefficients	13	10	5	7	3	8	2	2
20 coefficients	41	3	5	1	0	0	0	0
40 coefficients	44	2	3	1	0	0	0	0
No Mutation								
2 coefficients	1	4	11	12	9	7	3	3
3 coefficients	1	11	10	15	6	2	4	1
20 coefficients	19	23	7	1	0	0	0	0
40 coefficients	29	16	4	1	0	0	0	0

*Mutation is at 0.005 per locus per generation.

**Mutation from 0 to 1 and 1 to 0 each at 0.005 per locus per generation.

In summary the top two and bottom two lines of each block of Tables 1 and 2 appear to be qualitiatively different. We were led to examine in some detail the characteristics of those landscapes

that produced the highest numbers toward the right sides of Tables 1 and 2 when $n = 2$ or 3.

All of the cases in which high recombination was favored were characterized by high fitness values for the lowest and highest phenotypes, low values in the intermediate range, and a single minimum close to the center of the phenotypic range. One function of this form is the inverse Gaussian of the form

$$F(\nu) = k - \exp(\nu - \mu)^2/2\sigma^2. \tag{3}$$

Selection functions of this form can be viewed as *disruptive* in that they favor the phenotypic extremes at the expense of the intermediate values. Two versions of (3) were examined, each with "standard deviation" $\sigma = 33$. In one case we set $k = 2$ and in the other $k = 1$. "Mean" parameters $\mu = 20$ and 25 were used and the initial phenotypic value was $\bar{\nu} = 20$. Below are recorded the percentage of runs (out of 500) in which the high recombination allele CH rose to 100 percent from an initial frequency of 5 percent.

	$k = 1$	$k = 2$
$\mu = 20$	8.4%	4.4%
$\mu = 25$	9.6%	5.2%

(4)

The values 8.4 percent and 9.6 percent are significantly different from the 5 percent expected under neutrality. Starting from phenotypic mean values $\bar{\nu} = 10$ and 30 with both $\mu = 20$ and $\mu = 25$, however, there was no significant departure from neutrality in 500 runs. It would appear, then, that there is a delicate balance between the initial conditions in the population and the strength of disruptive selection insofar as evolution at the recombination locus is concerned.

Three other models that might be interpreted as disruptive selection were investigated in the same way. We chose $F(\nu) = |\nu - \mu|^m$ with $m = 1, 2, 4$, $\mu = 20$ and 25 and the initial population mean phenotype $\bar{\nu} = 20$. The results corresponding to the inverse Gaussian of the previous paragraph are below:

	$\lvert\nu - \mu\rvert$	$(\nu - \mu)^2$	$(\nu - \mu)^4$
$\mu = 20$	6.6%	7.8%	8.2%
$\mu = 25$	7.0%	6.6%	10.2%

(5)

Apparently the stronger is the strength of the directional selection the more likely is fixation on high recombination, but there is an interaction with the initial population configuration. This suggested

that similar results might be found for the deterministic model of recombination modification with two loci under selection.

3.4 Deterministic Model with Disruptive Selection

Here we use the usual two-locus selection model, with a third locus that controls recombination. The numerical analysis follows the description given in Bergman and Feldman (1990) except in the form of the 4×4 fitness matrix. The modifier locus is neutral with respect to selection and, for simplicity, no interference is assumed between recombination events in the two intervals. The phenotypic values of the genotypes are 0, 1, 2, 3, 4, according to the number of 1's in the genotype under selection. The initial frequencies of the high and low recombination alleles are equal to 0.5 unless mentioned otherwise, and initially there is no linkage disequilibrium. The exact 8-chromosome system was iterated for 512 generations and the frequency of CH at that time was recorded.

The fitness matrices considered may be written in the form

	11	10	01	00
11	w_4	w_3	w_3	w_2
10	w_3	w_2	w_2	w_1
01	w_3	w_2	w_2	w_1
00	w_2	w_1	w_1	w_0

(6)

where identical entries reflect the dependence of viability on the phenotypic value. Three kinds of "disruptive" viability matrices were considered. The first is of the form $w_i = |i - \mu|^{1/2}$, the second is $w_i = (i - \mu)^2$ and the third $w_i = k - \exp[-(i - \mu)^2/2\sigma^2]$. The results for the square-root case are recorded in Tables 3 and 4 below. In Table 3, μ was set at 1 and the frequency of CH recorded as a function of the initial mean phenotypic value, $\bar{\nu}$, in the population. In Table 4, $\bar{\nu}$ was set at 1.4 and the frequency of CH recorded as a function of μ. In both tables 3, 4 CH was recessive to CL.

Table 3: Fitness $|i-\mu|^{1/2}$; $\mu = 1.0$

$\bar{\nu}$	0.38	0.60	0.80	1.0	1.2	1.4	1.6	1.8	1.98	2.0
frequ(CH)	0.4868	0.0180	0.2400	0.3903	0.5033	0.5080	0.5069	0.5038	0.5026	0.5032
x_{11}	0	0	0	1	1	1	1	1	1	1

$\bar{\nu} = 1.2$

Frequency of allele 1

1st locus	2nd locus	CH frequ
.5999	.001	0.5033
.59	.01	0.5018
.55	.05	0.4955
.5	.1	0.4881

Table 4: Fitness Curve $|i-\mu|^{1/2}$; $\bar{\nu} = 1.4$*

μ	1.0	1.05	1.12	1.13	1.14	1.33	1.34	1.37	1.4	1.8	2.0
frequ (CH)	0.4922	0.4890	0.4850	0.4844	0.4838	0.4661	0.4646	0.4593	0.4525	0.4965	0.5021
x_{11}	1	1	1	1	1	1	1	1	1	1	1

*First locus frequency of allele 1 is 0.3, Second locus 0.4.

Examination of Table 3 reveals that when $\bar{\nu}$ is initially low the population eventually fixes on the selected chromosome 00 while larger initial values of $\bar{\nu}$ lead to fixation on 11 . The correspondence between this change with $\bar{\nu}$ and the allele favored at the recombination modifying locus is not perfect but is quite marked. Clearly, when $\bar{\nu}$ is initially greater than $\mu = 1.0$ by a large enough amount CH

is favored, otherwise CL is favored. It is not just the distance $|\bar{\nu} - \mu|$ that is important here as can be seen in Table 4, where $\bar{\nu}$ is fixed at 1.4. When $\bar{\nu}$ is sufficiently greater than μ, CH is favored but with $\mu < \bar{\nu}$ and $\bar{\nu} - \mu$ small enough CL is favored. In Table 4 we see again the relationship between fixation on the advantageous chromosome 11 and increase of CH.

In Table 5 are recorded the frequencies of CH after 512 generations when selection is parabolic. Two values of μ are illustrated and there are clear parallels between them. As $\bar{\nu}$ increases CH starts out at a disadvantage, becomes advantageous and then loses its advantage. We address this non-monotonicity in the Discussion.

Table 5. Parabolic Fitness $(i - \mu)^2$.

Frequency of allele 1				
1st locus	2nd locus	$\bar{\nu}$	$\mu = 0.4$	$\mu = 0.6$
0.05	0.1	0.3	0.4978	0.4755
0.04	0.16	0.4	0.5004	0.4855
0.1	0.15	0.5	0.5011	0.4926
0.1	0.2	0.6	0.5016	0.4959
0.15	0.25	0.8	0.5017	0.4990
0.2	0.3	1.0	0.5015	0.5000
0.4	0.6	2.0	0.5004	0.5003
0.5	0.7	2.4	0.5002	0.5002
0.7	0.8	3.0	0.5001	0.5001
0.9	0.9	3.6	0.5000	0.5000

For the third two-locus selection scheme, the inverse Gaussian, we set

$$w_i = k - \exp[-(i - \mu)^2/2\sigma^2] \tag{7}$$

where i takes the values $i = 0, 1, 2, 3, 4$, in the fitness matrix. With initial frequency 0.06 of allele 1 at the first selected locus, and 0.14 of allele 1 at the second selected locus, so that with linkage equilibrium $\bar{\nu}$ is 0.4 initially, we found that when $k = 1$, $\sigma = 1$ the fate of CH depended on μ. With

$\mu = 1, 2, 3$ the frequency of CH approached 0.4750, 0.4999, and 0.5005, respectively. Tables 6 and 7 show how the frequency achieved by CH after 512 generations, starting from 0.5, may depend on the initial value of μ, and σ, respectively. It is clear that all three of $\bar{\nu}, \mu, \sigma$ may influence whether CH eventually achieves a frequency higher than its initial value.

Table 6. Inverse Gaussian with $k = 1$, $\sigma = 0.5$, $\bar{\nu} = 2.0$*.

Frequency of CH is recorded.

μ	2.5	2.6	2.7	2.75	3
frequ (CH)	0.4977	0.4996	0.5022	0.5037	0.5107

*Each locus begins with the frequency of allele 1 at 0.5.

Table 7. Inverse Gaussian with $k = 2$, $\mu = 3$, $\bar{\nu} = 2$*.

Frequency of CH is recorded.

σ	frequ CH
0.5	0.5086
0.8	0.5054
0.9	0.5035
1.0	0.5019
1.05	0.5013
1.1	0.5007
1.15	0.5002
1.2	0.4997
1.25	0.4993
1.5	0.4975
2.0	0.4949

*Frequency of allele 1 at each locus is initially 0.5.

Dominant and Recessive Modifiers: In the previous numerical experiments the high recombination allele CH was recessive to CL. Does the recessivity of CH affect its evolution? The answer is in the affirmative as Table 8 shows. We set $\mu = 3$, $\sigma = 1$ in the inverse Gaussian and examined the fate of CH, after 512 generations, as a function of $\bar{\nu}$. Clearly in the dominant case CL is slightly favored while in the recessive case there may be dependence on the initial value of $\bar{\nu}$, with CH favored when $\mu - \bar{\nu}$ is large enough, and CL favored when $\mu - \bar{\nu}$ is small enough. In the third column of Table 8 the non-monotonicity is not an artifact; the frequency of CH after 512 generations is indeed higher when $\bar{\nu} = 1.8$ than elsewhere in the given range.

Table 8. Recessive and Dominant Cases with Inverse Gaussian.

$\mu = 3, \sigma = 1$

	k = 2		k = 1	
$\bar{\nu}$	Recessive	Dominant	Recessive	Dominant
1.6	0.5031	0.4997	0.5017	0.4999
1.8	0.5029	0.4997	0.5018	0.4999
1.98	0.5021	0.4997	0.5017	0.4999
2.0	0.5019	0.4997	0.5017	0.4999
2.2	0.4998	0.4996	0.5013	0.4999
2.4	0.4953	0.4993	0.5001	0.4998

*Each locus begins with frequency of allele 1 at 0.5.

A second set of comparisons between the dominant and recessive cases was based on the role of the initial average recombination fraction in the population. If we denote the initial frequencies of CH in the chromosomes carrying 00, 01, 10, and 11 by p, q, r, s, respectively, then the role of the initial average value of the recombination fraction in the population, $\bar{r}$, may be investigated as in Table 9. In this table, where the fitnesses are inverse Gaussian, the frequencies of CH after 512 generations are recorded. We see again that CH has lost ground to CL when the former is dominant, and gained when it is recessive, as we observed in Table 8. In our earlier study (Bergman and Feldman, 1990) we saw

a similar quantitative difference between the recessive and dominant cases, but did not find examples where high recombination advanced above its initial frequency when recessive, and lost ground when dominant. Notice also in Table 9 that this qualitative finding does not depend on $\bar{r}$.

Table 9. Dominant and Recessive Evolution as a Function of Initial Recombination Rate*: Inverse Gaussian Case with $\mu = 3$, $\sigma = 1$, $k = 2$, and $\bar{\nu} = 2$.**

$\bar{r}$	Recessive	Dominant†
0.01	0.0100	0.0100−
0.10	0.1002	0.0997
0.20	0.2006	0.1996
0.30	0.3011	0.2996
0.40	0.4015	0.3996
0.50	0.5019	0.4997
0.60	0.6021	0.5998
0.70	0.7021	0.6998
0.80	0.8017	0.7999
0.90	0.9015	0.8999
0.99	0.9901	0.9900−

* $\bar{r} = p = q = r = s$ is the initial frequency of CH.

** Each locus begins with frequency of allele 1 at 0.5.

†The negative sign following the number indicates that the actual frequency is *less* than that shown by about 10^{-5}.

Our final series of numerical studies involved variation in p, q, r, s. By choosing these to be different, while holding the initial frequencies of 00, 01, 10, 11 equal to 0.25, we establish linkage disequilibrium between the modifier locus and the selected genes with the selected genes initially in linkage equilibrium. Table 10 records a sample of results of this kind with inverse Gaussian fitnesses; $\mu = 3$, $\sigma = 1$,

$k = 2$, and $\bar{\nu} = 2$. Again the size of the shift in the frequency of CH is smaller when it is dominant than in the recessive case. The initial distribution of CH among the selected chromosomes is seen to have a strong effect in the four examples with $\bar{r} = (p + q + r + s)/4 = 0.55$. In the first case CH increases to well above its initial frequency at 512 generations, but in the other three examples it decreases sharply. The last two examples should be compared to the cases $\bar{r} = 0.7$ and 0.8 of Table 9, where CH gains in the recessive case. In Table 10, CH loses when $\bar{r} = 0.75$ and 0.775, presumably an effect of the initial disequilibrium.

Table 10. The Effect of the Initial Distribution of CH:

Inverse Gaussian Selection with $\mu = 3$, $\sigma = 1$, $k = 2$, $\bar{\nu} = 2$**.

							Linkage Disequilibria*			
$\bar{r}$	p	q	r	s	Recessive	Dominant	D_{12}	D_{13}	D_{23}	D_{123}
0.15	0.2	0.1	0.2	0.1	0.198	0.164	0	0	0.025	0
0.25	0.3	0.4	0.1	0.2	0.292	0.257	0	0.05	−0.025	0
0.25	0.3	0.2	0.4	0.1	0.307	0.275	0	0	0.05	−0.0125
0.475	0.6	0.7	0.2	0.4	0.581	0.498	0	0.0875	−0.0375	0.00625
0.55	0.7	0.4	0.6	0.5	0.687	0.584	0	0	0.05	0.0125
0.55	0.4	0.2	0.9	0.7	0.409	0.521	0	−0.125	0.05	0
0.55	0.2	0.9	0.6	0.5	0.278	0.488	0	0	−0.075	−0.05
0.55	0.1	0.9	0.3	0.9	0.150	0.428	0	−0.025	0.175	−0.0125
0.75	0.7	0.5	0.9	0.9	0.700	0.734	0	−0.075	0.025	0.0125
0.775	0.7	0.9	0.8	0.7	0.717	0.767	0	0.0125	−0.0125	−0.01875

*Linkage Disequilibria computed as e.g. in Feldman et al. (1974).

**Each locus begins with frequency of allele 1 at 0.5.

3.5 Recombination Dynamics: Conclusions

The mapping from genotype to fitness in a diploid model with 20 genes could be extremely complicated. In our first models with finite population size we have simplified this relationship enormously by interposing a simple mapping from genotype to phenotype, known in quantitative genetics *as additive determination of the phenotype*. By adding the number of 1's in the genotype the domain of the fitness function is greatly restricted. While the fitness mapping (Eq.1) used here is reasonably complicated by the standards of population genetics, it is certainly much simpler than those used by Tanese (1989) and Forrest and Mitchell (1991), which were defined by Walsh polynomials. Nevertheless, Tables 1 and 2 show clearly that as the fitness mapping becomes more complicated (i.e. n increases) the likelihood that a high recombination allele succeeds drops sharply.

In these finite population models with 2 or 3 coefficients the shapes of the fitness mappings that favored high recombination were all of the *disruptive* kind, that is the extreme phenotypes had the highest and the intermediate phenotypes the lowest fitnesses. As can be seen in the results (Eq. 4) the *strength* of the disruptive selection may play a critical role; with $k = 1$ in (Eq. 3) the inverse Gaussian favors the extremes more sharply than when $k = 2$. The results (Eq. 5) for polynomial selection appear to reinforce the idea that the stronger the disruptive selection, the more likely is high recombination to succeed. A caveat should be made: in these finite population studies we did not make a detailed survey of the initial distribution of chromosomes nor of the role of the initial value of $\bar{\nu}$. On the basis of the deterministic results for 2 genes it is reasonable to conjecture that both may play a role in the ultimate fate of a high-recombination allele.

For the deterministic two-locus model, Tables 3-6 and Table 8 amply document the role of the initial average phenotype $\bar{\nu}$ and the distribution of chromosomes that produce this average. The ultimate frequency of CH is not monotonic as a function of $|\bar{\nu} - \mu|$, and for the same value of $\bar{\nu}$, different initial distributions of the alleles at the first and second loci (with initial linkage equilibrium) may lead to different outcomes for high recombination (Table 3 for example). Also of interest here is the shape of the disruptive selection function; in the universe Gaussian case of Table 6, as $\mu - \bar{\nu}$ increased the ultimate frequency of CH also increased, while with the paraboloid fitnesses of Table 5 there was first an increase, but for the largest values of $\bar{\nu} - \mu$ the trend reversed. The strength of the

selection in the inverse Gaussian case is measured by σ^2 for fixed k. Table 7 shows that the greater the advantage to the extremes (the smaller is σ), the greater is the advantage to CH.

In our previous study of directional Gaussian selection there was a tendency for CH to change less and more slowly when it was dominant than when recessive. On the whole, however, the direction of change as the same in both cases. Table 8 shows that with inverse Gaussian selection the first observation is still valid but the second is not. It appears much more difficult for CH to advance in the dominant case, irrespective of the initial recombination value in the population (Table 9).

Table 10 reveals a phenomenon that appears to be new. The linkage disequilibrium between the major loci is initially zero, yet the fate of CH depends delicately on the initial recombination pattern in the population. For example, when the average recombination rate is 0.55, CH may advance sharply or drop sharply depending on the exact distribution of CH and CL among the selected chromosomes. Again when CH is dominant this effect is more muted. From Table 10 it is difficult to discern a constant pattern for the effect of the other linkage disequilibria that might explain this finding.

These delicate dependencies on the shape and strength of the disruptive selection, on the initial average phenotype and its distribution, and on the distribution of CH among the selected chromosomes conspire to make generalizations very difficult. Perhaps the only general conclusion we may draw is that when disruptive selection is strong, there will be a set of initial chromosome frequency vectors in the population from which evolution will favor CH. On the other hand, under the same conditions CL will usually be favored for some other set of starting conditions. As selection becomes stronger, the latter set appears to decrease in size relative to the former.

4 Future Work

In the course of our research on the use of biologically-inspired computational pradigms for signal processing problems, numerous questions have arisen. In this report we will describe our general approach in this research and our plan for future work. In particular, we will point out the relationship of our work to Kohonen's feature-map networks and the ways in which we propose to generalize and expand this area. The second research area is the evolution of learning and plasticity from a population genetics view point. The third area of proposed research is an expansion of the evolution computational paradigm to include coevolutionary processes.

4.1 Introduction

Recently, many researchers have speculated on the possible relationships between neural networks and genetic algorithms. One reason for this interest, no doubt, is because both concepts are derived from fundamentally biological metaphors, so it is natural to consider them in combination. More significantly, genetic algorithms suggest new ways to construct optimal neural networks that avoid some serious problems associated with conventional learning algorithms — in particular, the problems of slow learning, local error minima, poor generalization, and the need for large training sets.

We shall argue that the use of genetic algorithms primarily for *optimization* is based on a somewhat naive view of biological evolution and that it neglects several important features of the metaphor. In the modern neo-Darwinian view, evolution (in the biological sphere) does not produce optimal individual organisms in any well-defined sense. Instead, it appears to produce well-adapted ecosystems, with subpopulations of genetically related individuals inhabiting their own niches, but interacting in complex and often unpredictable ways. Biological evolution is characterized by increasing diversity — as relatively small, undiversified parent populations radiate into newly uncovered niches, the gene pool splits into a more diverse collection of species. This picture contrasts quite starkly with the "evolution as optimization" point of view, in which the ideal end result is a population of identical, optimal clones.

Replacing "optimization" with "adaptation" actually suggests a much richer and potentially more

effective role for genetic algorithms. Instead of an optimal neural network to solve a relatively simple static problem, a truly adaptive system may consist of a diversified collection of networks that specialize in subproblems that are part of a more complex, dynamic, and possibly ill-defined problem.

In the following section we briefly present some relevant background material, including a more precise definition of adaptation. Next we discuss our previous work on this problem, and finally a plan for further research.

4.2 Background

In our previous work we have used the genetic algorithm approach to create the computational paradigm for signal processing problems. The following steps are used in our Genetic Algorithm approarch:

- A problem is selected and a class of computational mechanisms thought to be effective for solving the problem is identified.
- A coding scheme is devised for specifying members of the class of mechanisms.
- A population of encodings and associated mechanisms is constructed.
- The mechanisms are tested on instances of the problem and are graded according to their performance. Those with higher grades are considered to be more "fit."
- A new population is constructed by selecting the most fit mechanisms, producing a new set of mechanisms by combining the encodings of the more fit one, and inserting them into the population.
- This process is repeated until the population becomes dominated by (one hopes) optimally fit mechanisms. The most fit is taken to be the best solution to the problem.

The genetic-algorithm approach is obviously inspired by the phenomenon of biological evolution. We argue that the simple approach outlined above is deficient in several respects.

1. Biological evolution does *not* optimize; it adapts.

2. There is no objective "fitness function."

3. Significant problems may not be solvable by a *single* mechanism. Instead, the solution may require a collection of individuals, or perhaps subpopulations of individuals, specialized to different subproblems.

4. The implication that the problem to be solved is static often is not realistic, and certainly does not follow the example of biological environments. Instead, we argue, the flexibility of biological evolution in the face of changing environments suggests that genetic algorithms may best suited to problems that change over time.

5. The role of development may be crucial. It is clear that in the course of transforming from the organism's genotypic description to its phenotype, an organism goes through numerous layers of developmental stages by which each layer creates an ever more complex entity. While in the experiments we have conducted this transformation is shallow, yet an interesting behavior can be observed; a more logically deep transformation will create better framework for hierarchically organized systems.

4.3 Open Questions

During the last year several important questions about the potential of our approach and relationship of our ideas to different areas of neural network and dynamical system science have surfaced.

- When the evolutionary system is constrained to work within a framework that allows only local reproduction (vs. global reproduction), there is a question as to whether one can relate the "map-like" activity of the system to the feature maps generated by Kohonen's self organization network. The answer to this question is "yes," at least with regard to islands of activity, namely, sources that are considered to come from the same generator. Those required to be recognized by the same "individuals" are clustered together. However nearby signals (in signal space) coming from different generators may be separated spatially.

- One observed characteristic of our system is the emergence of *species*, networks that are better at recognizing stimuli that are part of one environment and not the other. This emergent behavior

is achieved by the limited capability each network is endowed with. It is also a consequence of the fact that the individuals are temporally isolated during the reproduction process; namely, at each time step only the top-N, the individuals that are best fit at recognizing a signal are mated. This process is one way to achieve reproductive isolation, a required condition for speciation. The questions we would like to address are:

1. What are the minimal conditions for the process of speciation to occur?
2. What are the advantages of having a system with several species as opposed to a system with one "optimized" individual?
3. Is it possible that learning capabilities will evolve and under what conditions. We will study this question from a population genetic point of view. The question to ask is: Under what environmental conditions learning capability evolve, and what complexity of learning mechanism does an individual need in order to cope with a known environmental complexity?
4. Is it possible to use an evolutionary-inspired system to capture the behavior of a dynamical system by reflecting in its behavior the nature of the dynamical system, e.g. can one regenerate attractors that are associated with the "true" attractors used in training the system?

In what follows, we will address some of the issues in more detail and describe some of the preliminary results obtained by simulations.

4.4 Relation to Feature-Map Networks

Our evolutionary approach is related to self-organizing feature maps by neural networks (Kohonen 83). Such maps are of interest because of their reduced data dimensionality capability, since economic representation of data with all their interrelationships is a crucial problem in information sciences. The ability to reduce dimensionality by forming a *reduced representation* of the most relevant facts, without loss of knowledge about their interrelationships, is a desirable characteristic one would like to achieve. When the reduction in dimensionality takes place, certain *geometric* relationships should

be maintained, i.e. nearby objects in the higher-dimensional space (can be an n-dimensional feature vector) should be mapped into nearby objects in the reduced space. In Kohonen's network, an n-dimensional feature vector is mapped into *geographicaly* close units (organized on a two-dimensional grid).

To get the self-organization of the feature maps, two processes of lateral feedback should take place: the first is local excitation, where each unit which is activated excites units in its neighborhood. The second process is the inhibition of all units that are outside the excitation neighborhood. Once the system relaxes, the most activated unit(s) and its neighbors adjust their weights to maximize their response for the current feature vector. The interaction between the two processes creates a geometrically organized feature map of the knowledge in a particular category. That is to say, objects belonging to the same or similar categories will be represented by geometrically close units on the reduced-dimensionality feature map.

As we demonstrated in our previous work (in preparation), our system of a population of networks can exhibit similar behavior. The link between the feature-map network and the competing network system can be viewed as follows. The two lateral feedback processes, local excitation and global inhibition, can be associated with local mating and global selection respectively.

Local mating causes nearby processes (networks) to respond to closely related input vectors (or feature vector in general). This association comes about because the mating process mixes the properties of the two networks that mate. Such a process, after some generations, creates a system where geographically close networks share common properties, e.g., a similar set of weights, such that they response strongly to similar stimuli.

The process of differential reproduction due to selection is equivalent to the process of global inhibition: namely, only few members of the population are allowed to reproduce and enhance their response to the current input.

To summarize, one can view the system of competing networks as an extention of Kohonen's self-organized feature map network, where each unit is replaced by a general process that may itself be a neural network. Each network is capable of solving part of the problem, and the system as a unit solves the larger problem. In case the problem is within the complexity that a single network can

solve, it may be the case that each individual of the population will solve the entire problem.

4.5 Evolution of Learning: A Population Genetics Approach

The evolution of learning capabilities in organisms is one of the more perplexing issues in evolutionary biology. Several studies on the evolution of learning proposed the idea of learning as a mechanism to adapt to changes in the environment during somatic time. These studies are based on the "absolute fixity argument"; that is to say, in the presence of an absolutely fixed environment, an individual should develop a genetically fixed pattern of behavior (assuming some cost associated with learning).

On the other hand, in an absolutely unpredictable environment, where the past and present state of the environment bears no information about the future, then there is nothing to learn, and assuming some cost to learning, there is no driving force for learning capabilities to evolve.

Stephens (personal communication) proposed a different approach. Stephens argues that the pattern of predictability in relation to an individual's life history determines the evolution of learning. His study concludes that the value of learning is for those things that change between-generations and are regular within-generations.

An alternative approach is to view learning as the ability of an individual to construct a correct model of its environment and by proper use of the model to be able to predict future states of its environment.

Consider a changing environment where the state, $s \in \{0; 1\}$, is a stationary first-order Markov process, $\mathcal{S}$. The state of the environment at $t+1$ depends only on the state at time t, that is to say, that the conditional probability $P(s^{t+1} \mid \{s^t, s^{t-1}, ..., s^0\})$ depends only on s^t and is independent of t.

Each environmental state has a viability value, E_{s^t}, associated with it. In the current model the viabilities are: $E_0 = 0$ for $s^t = 0$ and $E_1 = 1$ for $s^t = 1$.

Consider a diallelic two-loci diploid model where the first locus (considered the main gene) controls the capability of an individual to learn and the second gene is a modifier gene which controls the probability of expressing the learning capability. If learning is expressed, an individual pays a cost $0 < \epsilon < 1$.

Individuals, either lacking or not expressing their learning capability, always sample from the environment regardless of its previous state.

Individuals that expresses their learning capabilities are endowed with a variable size "lookup" table containing the block probabilities of sampling from the environment. If an individual chooses to sample from the environment, its viability is increased proportional to the viability associated with the environment at the sampling time. If an individual chooses not to sample, its viability is increased by some factor $0 < \delta < 1$.

Learning is viewed as a two-step process; first, update a lookup-table based of observations, namely, generate an estimate, $p(s_i^{t+1} \mid s_j^t)$, of the true environmental conditional probabilities, $P(s_i^{t+1} \mid s_j^t)$. Second, once an individual is endowed with the model, it makes use of it for N time steps.

Several questions can be asked:

1. What environmental conditions, namely, what range of values $P(s_i^{t+1} \mid s_j^t)$ can take, will lead to the invasion of the learning individuals?

2. Will an individuals evolve to have a larger lookup-table for environments modeled as K's ordered Markov process?

3. In case the environment is not a Markov process, is it possible to evolve a more efficient learning mechanism other than a lookup-table?

4. What is the number of training steps each individual should go through to get optimal resalts for a given environment?

4.6 Proposed Work

The future work is a continuation and elaboration of our existing ongoing project on an evolutionary approach to learning.

In the future we would like to investigate three research areas: two processes described in the previous section and the third described in the body of the report. First we would like to attain a better and formal understanding of the relation between the feature maps generated by Kohonen's network and the generalization of the system we have been investigating. A detailed outline of the approach will

be discussed later. This work will be tightly linked to the investigation of dimensionality reduction, where the dimensions under consideration are the geometrical organization of the individuals in the population.

The second area of research will be on the evolution of learning capabilities. This research will lead to a better undesrstanding of the conditions under which learning mechanism as opposed to fix algorithm is advantagous. It will reflect also on the question of what should be the number of learning steps before performing a genetic operation like recombination and mutation.

The third proposed direction is the investigation of the effect of coevolutionary processes on the formation of clusters in the population and maintaining variability in a controlled way to preserve memory of past experience in the presence of a changing environment.

The results of the research will lead to better understanding of the relationship among neural network theory, evolutionary and population genetics, and some aspects of dynamical systems theory. We expect also that fields such as signal processing and machine learning will greatly benefit from the outcome of this research.

5 Publications and Presentations

A paper by Stephen T. Barnard and Aviv Bergman has been published in the *Proceedings of Parallel Problem Solving from Nature*, a workshop held in Germany in October 1990. Aviv Bergman also participated in the international workshop on ***Evolution and Complex Systems***, in Torino, Italy, in July 1990. This workshop included fruitful discussion among several of the world's top researchers in complex systems and evolution.

A second paper by Aviv Bergman and Marcus W. Feldman will be published in *Physica D*, Recombination Dynamics and the Fitness Landscape.

A third paper by Aviv Bergman, Means of Variability, is in preperation.

All three paperes have been presented by Aviv Bergman at the Santa Fe Institute during the summer of 1991 as part of their Adaptive Computation program.

References

[1] Ackley, D. H., G. E. Hinton, and T. J. Sejnowski, "A learning machine for Boltzman Machines," *Cognitive Science, 9*, 147-169, 1985.

[2] Bergman, A., and M.W. Feldman, "More on selection for and against recombination," *Journal for Theoretical Population Biology*, 1990.

[3] Bergman, A., and M. Kerszberg, "Breeding intelligent automata," *IEEE First Annual Conference on Neural Networks*, San Diego, June 1987.

[4] Ewens, W.J., *Mathematical Population Genetics*, Springer-Verlag, 1979.

[5] Goldberg, David E., *Genetic Algorithms*, Addison-Wesley, 1989.

[6] Hartl, D.L., and A.G. Clark, *Principles of Population Genetics*, (2nd edition), Sinauer Associates, Inc. Publishers, 1989.

[7] Holland, J.H., *Adaptation in Natural and Artificial Systems*, Ann Arbor : University of Michigan Press, 1975.

[8] Kerszberg, M., and A. Bergman, "The evolution of data processing abilities in competing automata," in *Computer Simulation in Brain Science*, ed. Rodney M.J.Cotterill, pp. 249-259, Cambridge, U.K.: Cambridge University Press, 1988.

[9] Packard, N., "Evolving bugs in a simulated ecosystem," in *Artificial Life*, ed. Christopher G. Langton, Addison Wesley Publishing Company, 1989.

[10] Roughgarden, J., *Theory of Population Genetics and Evolutionary Ecology: An Introduction*, Macmillan Publication Co., Inc., 1979.

[11] Rumelhart, D. E., G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing*, vol. 1, ed. D. E. Rumelhart and J. L. McClelland, Cambridge, Massachusetts: The MIT Press 1986.

[12] Bateson, P.P.G., "Gene, evolution, and learning," in *The Biology of Learning*, eds. P. Marler and H.S. Terrace, pp. 75-88: Dahlem Konferenzen 1984, Springer-Verlag, 1984.

[13] Changeux, J.P., T. Heidmann, and P. Patte, "Learning by selection," in *The Biology of Learning*, eds. P. Marler and H.S. Terrace, pp. 115-133: Dahlem Konferenzen 1984, Springer-Verlag 1984.

[14] Gold, J.L. and P. Marler, "Ethology and the natural history of learning," in *The Biology of Learning*, eds. P. Marler and H.S. Terrace, pp. 47-74: Dahlem Konferenzen 1984, Springer-Verlag 1984.

[15] Maxwell, T., C. Lee Giles, and Y. C. Lee, "Generalization in neural networks: The contiguity problem," *Proceedings of the IEEE International Conference on Neural Networks*, San Diego, CA, June 1987.

[16] Edelman, G.M., "Group selection and phasic reentrant signaling: A theory of higher brain function," in *The Mindful Brain* (with V. B. Mountcastle), Cambridge, Massachusetts: M.I.T. Press, 1987.

[17] Reeke, G.N, and G.M. Edelman, "Selective networks and recognition automata," *Annals of the New York Academy of Science,* Vol. 429, 181-201, 1984.

[18] Bergman, A. and M.W. Feldman. "More on selection for and against recombination." *Theor. Pop. Biol.* **38**: 68–92. 1990.

[19] Charlesworth, B. "Recombination modification in a fluctuating environment." *Genetics* **83**: 181–195. 1976.

[20] Charlesworth, D., B. Charlesworth, and C. Strobeck. "Selection for recombination in partially self-fertilizing populations." *Genetics* **93**: 237–244. 1979

[21] Feldman, M.W., F.B. Christiansen, and L.D. Brooks. "Evolution of recombination in a constant environment." *Proc. Natl. Acad. Sci. USA* **77**: 4838–4841, 1980.

[22] Feldman, M.W., I. Franklin, and G.J. Thomson. "Selection in complex genetic systems I. The symmetric equilibria of the three-locus symmetric viability model." *Genetics* **76**: 135–162, 1974.

[23] Feldman, M.W. and U. Liberman. " An evolutionary reduction principle for genetic modifiers." *Proc. Natl. Acad. Sci. USA* **83**: 4824–4827, 1986.

[24] Feldman, M.W. and S.P. Otto. "More on recombination and selection in the modifier theory of sex-ratio distortion." *Theor. Pop. Biol.* **35**: 207–225, 1989.

[25] Forrest, S. and M. Mitchell, "The performance of genetic algorithms on Walsh polynomials: Some anomalous results and their explanation." *International Conference on Genetic Algorithms*, San Diego, 1991.

[26] Holsinger, K.E. and M.W. Feldman. "Linkage modification with mixed random mating and selfing: A numerical study," *Genetics* **103**: 323–333, 1983.

[27] Holsinger, K.E., M.W. Feldman, L. Altenberg. "Selection for increased mutation rates with fertility differences between matings," *Genetics* **112** 909–922, 1986.

[28] Kaufman, S., "Towards a general theory of adaptive walks on rugged landscapes," *J. Theor. Biol.* **128**: 11–45, 1987.

[29] Levene, H. "Genetic equilibria when more than one ecological niche is available," *Amer. Natur.* **87**: 331-333, 1953.

[30] Liberman, U. and M.W. Feldman. "Modifiers of mutation rate: A general reduction principle," *Theor. Pop. Biol.* **30**: 341–371, 1986.

[31] Liberman, U. and M.W. Feldman. "A general reduction principle for genetic modifiers of recombination," *Theor. Pop. Biol.* **30**: 341–371, 1986.

[32] Liberman, U. and M.W. Feldman. "The Reduction Principle for Genetic Modifiers of the Migration Rate," pp. 111–137. *In* Feldman, M.W. (ed.), *Mathematical Evolutionary Theory*. Princeton University Press, Princeton, N.J., 1989.

[33] Maynard Smith, J. "The effects of normalizing and disruptive selection on genes for recombination," *Genet. Res.* **33**: 121–128, 1979.

[34] Maynard Smith, J. "Selection for recombination in a polygenic model," *Genet. Res.* **35**: 269–277, 1980.

[35] Maynard Smith, J. "Selection for recombination in a polygenic model - The mechanism," *Genet. Res.* **51**: 59–63, 1988.

[36] Tanese, R. "Distributed genetic algorithms for function optimization." Ph.D. Dissertation. University of Michigan, 1989.

[37] Thomson, G. and M.W. Feldman. "Population genetics of modifiers of meiotic drive. II. Linkage modification in the segregation distortion system," *Theor. Pop. Biol.* **5**: 155–162, 1974.

[38] Twomey, M.J. and M.W. Feldman. "Mutation modification with multiplicative fertility selection," *Theor. Pop. Biol.* **37**: 320–342, 1990